Date=28/10/2020
Lecture By=Manish Mahant
Subject ⇒Ajax & HTTP Requests

IN PREVIOUS LECTURE **(QUICK RECAP) Date-27/10/2020**	**In Today's Lecture (Overview)**
What is functions Defining functions Calling functions Function scope What is Jquery Why jQuery? jQuery Syntax Questions For Self Practice // CC For the Day	AJAX AJAX Example Explained HTML Page What is AJAX? HTTP Methods The GET Method The POST Method The PUT Method The HEAD Method The DELETE Method The OPTIONS Method Compare GET vs. POST Questions For Self Practice // Assignment For The Day

AJAX

AJAX is a developer's dream, because you can:

- Read data from a web server - after a web page has loaded
- Update a web page without reloading the page
- Send data to a web server - in the background

AJAX Example Explained

HTML Page

```
<!DOCTYPE html>

<html>

<body>

<div id="demo">

  <h2>Let AJAX change this text</h2>

  <button type="button" onclick="loadDoc()">Change Content</button>

</div>

</body>

</html>
```

The HTML page contains a section and a <button>.

The section is used to display information from a server.

The <button> calls a function (if it is clicked).

The function requests data from a web server and displays it:

What is AJAX?

AJAX = Asynchronous JavaScript And XML.

AJAX is not a programming language.

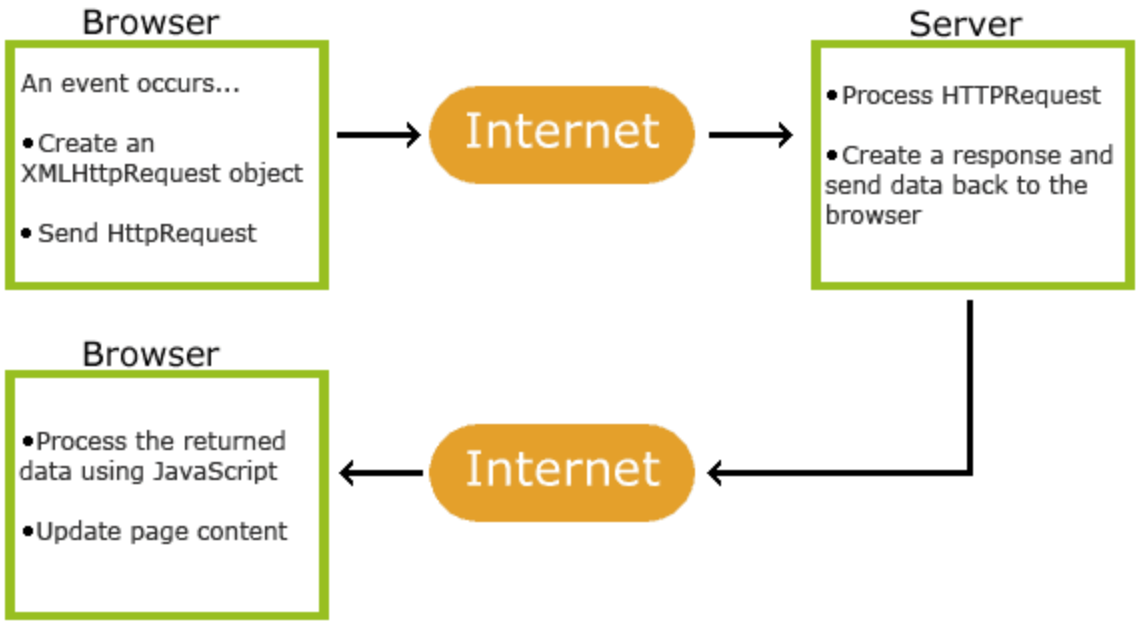
AJAX just uses a combination of:

- A browser built-in XMLHttpRequest object (to request data from a web server)
- JavaScript and HTML DOM (to display or use the data)

- 1. An event occurs in a web page (the page is loaded, a button is clicked)
- 2. An XMLHttpRequest object is created by JavaScript
- 3. The XMLHttpRequest object sends a request to a web server
- 4. The server processes the request
- 5. The server sends a response back to the web page
- 6. The response is read by JavaScript

- 7. Proper action (like page update) is performed by JavaScript

---



HTTP Methods

- GET
- POST
- PUT
- HEAD

- DELETE
- PATCH
- OPTIONS

The GET Method

GET is used to request data from a specified resource.

GET is one of the most common HTTP methods.

Note that the query string (name/value pairs) is sent in the URL of a GET request:

```
/test/demo_form.php?name1=value1&name2=value2
```

Some other notes on GET requests:

- GET requests can be cached
- GET requests remain in the browser history
- GET requests can be bookmarked
- GET requests should never be used when dealing with sensitive data
- GET requests have length restrictions
- GET requests are only used to request data (not modify)

The POST Method

POST is used to send data to a server to create/update a resource.

The data sent to the server with POST is stored in the request body of the HTTP request:

```
POST /test/demo_form.php HTTP/1.1
Host: w3schools.com
name1=value1&name2=value2
```

POST is one of the most common HTTP methods.

Some other notes on POST requests:

- POST requests are never cached

- POST requests do not remain in the browser history
- POST requests cannot be bookmarked
- POST requests have no restrictions on data length

The PUT Method

PUT is used to send data to a server to create/update a resource.

The difference between POST and PUT is that PUT requests are idempotent. That is, calling the same PUT request multiple times will always produce the same result. In contrast, calling a POST request repeatedly have side effects of creating the same resource multiple times.

The HEAD Method

HEAD is almost identical to GET, but without the response body.

In other words, if GET /users returns a list of users, then HEAD /users will make the same request but will not return the list of users.

HEAD requests are useful for checking what a GET request will return before actually making a GET request - like before downloading a large file or response body.

The DELETE Method

The DELETE method deletes the specified resource.

The OPTIONS Method

The OPTIONS method describes the communication options for the target resource.

Compare GET vs. POST

The following table compares the two HTTP methods: GET and POST.

	GET	**POST**
BACK button/Reload	Harmless	Data will be re-submitted (the browser should alert the user that the data are about to be re-submitted)
Bookmarked	Can be bookmarked	Cannot be bookmarked
Cached	Can be cached	Not cached
Encoding type	application/x-www-form-urlencoded	application/x-www-form-urlencoded or multipart/form-data. Use multipart encoding for binary data
History	Parameters remain in browser history	Parameters are not saved in browser history

Restrictions on data length	Yes, when sending data, the GET method adds the data to the URL; and the length of a URL is limited (maximum URL length is 2048 characters)	No restrictions
Restrictions on data type	Only ASCII characters allowed	No restrictions. Binary data is also allowed
Security	GET is less secure compared to POST because data sent is part of the URL Never use GET when sending passwords or other sensitive information!	POST is a little safer than GET because the parameters are not stored in browser history or in web server logs
Visibility	Data is visible to everyone in the URL	Data is not displayed in the URL

Questions For Self Practice // Assignment For The Day

https://au-assignment.s3.ap-south-1.amazonaws.com/Week_18_Day_3_Assignment-a64f2dd8-68cb-44e6-8033-1694b69fab42.pdf